**Phase 2 project**

**Project Title :** NOISE POLLUTION MONITORING

**Project ID** : proj\_223738\_Team\_6

**College :** Gnanamani College of Technology

**College code :** 6208

**Branch :** B.Tech/Information Technology

**Year :** IIIrd year

**Team Members:**

S.HariHaran (620821205018)

C.Barath (620821205007)

M.Deepak (620821205011)

S.M.Rohith(620821205046 )

V.Bhuvaneswaran(620821205008)

Noise pollution monitoring

**Additional implemention:**

An IoT-based noise pollution monitoring system is using a Class 1 Sound Level Meters,MEMS Microphones.These are highly accurate and compliant with international standards, making them suitable for precise noise measurements.MEMS microphones are compact, low-power, and capable of capturing a wide range of frequencies, making them valuable for continuous monitoring.Sensors is to monitor and quantify the levels of unwanted or harmful

We use **Smart sensor**. It is more sophisticated sensors are being developed that can measure not only noise levels but also types of sounds, such as traffic noise,

industrial noise, or wildlife sounds. This helps in identifying the sources of noise pollution more precisely.

**Components Needed:**

1. ESP8266Sound sensor

2. Arduino R3

3 .Microphone

4.Wi-Fi module

5.Batteries/solar panels

6.Databases/Cloud storage

7.LCD

8.GPS module

9.Geospatial software

10.Microphone windshield

11.MATT/HTTP

12.Antenna

**PHASE 2**

**Module 1: Data Collection &Data processing**

Continuously read audio data from the microphone sensor.

Sample the sound at regular intervals (e.g., every second).

Convert the analog sound signal to a digital format for processing.

Calculate the average sound level over a specified time period (e.g., 1 minute) by taking multiple samples and averaging them.

Compare the measured sound level with predefined noise thresholds to determine the noise level category (e.g., quiet, moderate, loud).

Store the processed data along with timestamps in a database or cloud storage for later analysis.

**Module 2: Real-time Monitoring**

Create a user-friendly interface, such as a web dashboard or mobile app, to display the current noise level and trends over time.

Update the interface in real-time to provide immediate feedback to users.

**Module 3: Alerts and Notifications**

Set up threshold-based alerts to notify users when noise levels exceed certain limits via email, SMS, or push notifications.

**Module 4: Data Analysis**

Perform data analysis to identify noise patterns, peak noise events, and trends over longer periods.

Generate reports and visualizations to help users understand noise pollution in their area.

**Module 5: User Interaction**

Allow users to customize alert thresholds and view historical noise data.

Provide options for users to share noise data on social media or with local authorities to raise awareness about noise pollution.

**Module 6: Data Privacy and Security**

Implement security measures to protect the collected data and user information